

**"Text-to-speech method and system, computer program  
product therefor"**

Field of the invention

5       The present invention relates to text-to-speech techniques, namely techniques that permit a written text to be transformed into an intelligible speech signal.

Description of the related art

10       Text-to-speech systems are known based on so-called "unit selection concatenative synthesis". This requires a database including pre-recorded sentences pronounced by mother-tongue speakers. The vocalic database is single-language in that all the sentences  
15 are written and pronounced in the speaker language.

      Text-to-speech systems of that kind may thus correctly "read" only a text written in the language of the speaker while any foreign words possibly included in the text could be pronounced in an intelligible way,  
20 only if included (together with their correct phonetization) in a lexicon provided as a support to the text-to-speech system. Consequently, multi lingual texts can be correctly read in such systems only by changing the speaker voice in the presence of a change  
25 in the language. This gives rise to a generally unpleasant effect, which is increasingly evident when the changes in the language occur at a high frequency and are generally of short duration.

      Additionally, a current speaker having to  
30 pronounce foreign words included in a text in his or her own language will be generally inclined to pronounce these words in a manner that may differ - also significantly - from the correct pronunciation of the same words when included in a complete text in the  
35 corresponding foreign language.

By way of example, a British or American speaker having to pronounce e.g. an Italian name or surname included in an English text will generally adopt a pronunciation quite different from the pronunciation adopted by a native Italian speaker in pronouncing the same name and surname. Correspondingly, an English-speaking subject listening to the same spoken text will generally find it easier to understand (at least approximately) the Italian name and surname if pronounced as expectedly "twisted" by an English speaker rather than if pronounced with the correct Italian pronunciation.

Similarly, pronouncing e.g. the name of a city in the UK or the United States included in an Italian text read by an Italian speaker by adopting the correct British English or American English pronunciation will be generally regarded as an undue sophistication and, as such, rejected in common usage.

The problem of reading a multi lingual text has been already tackled in the past by adopting essentially two different approaches.

On the one hand, attempts were made of producing multi lingual vocalic databases by resorting to bilingual or multi lingual speakers. Exemplary of such an approach is the article by C. Traber et al.: "From multilingual to polyglot speech synthesis" - Proceedings of the Eurospeech, pages 835-838, 1999.

This approach is based on assumptions (essentially, the availability of a multi-lingual speaker) that are difficult to encounter and to reproduce. Additionally, such an approach does not generally solve the problem generally associated to foreign words included in a text expected to be pronounced in a (possibly remarkably) different manner

from the correct pronunciation in the corresponding language.

Another approach is to adopt a transcriptor for a foreign language and the phonemes produced at its output which, in order to be pronounced, are mapped onto the phonemes of the languages of the speaker voice. Exemplary of this latter approach are the works by W.N. Campbell "Foreign-language speech synthesis" Proceedings ESCA/COCSDA ETRW on Speech Synthesis, Jenolan Caves, Australia, 1998 and "Talking Foreign. Concatenative Speech Synthesis and Language Barrier", Proceedings of the Eurospeech Scandinavia, pages 337 - 340, 2001.

The works by Campbell essentially aim at synthesizing a bilingual text, such as English and Japanese, based on a voice generated starting from a monolingual Japanese database. If the speaker voice is Japanese and the input text English, an English transcriptor is activated to produce English phonemes. A phonetic mapping module maps each English phoneme onto a corresponding, similar Japanese phoneme. The similarity is evaluated based on the phonetic - articulatory categories. Mapping is carried out by a searching a look-up table providing a correspondence between Japanese and English phonemes.

As a subsequent step, the various acoustic units intended to compose the reading by a Japanese voice are selected from the Japanese database based on their acoustic similarities with the signals generated when synthesizing the same text with an English voice.

The core of the method proposed by Campbell is a lookup-table expressing the correspondence between phonemes in the two languages. Such table is created manually by investigating the features of the two languages considered.

In principle, such an approach is applicable to any other pair of languages, but each language pair requires an explicit analysis of the correspondence therebetween. Such an approach is quite cumbersome, and  
5 in fact practically infeasible in the case of a synthesis system including more than two languages, since the number of language pairs to be taken into account will rapidly become very large.

Additionally, more than one speaker is generally  
10 used for each language, having at least slightly different phonologic systems. In order to put any speaker voice in a condition to speak all the languages available, a respective table would be required for each voice - language pair.

15 In the case of a synthesis system including N languages and M speaker voices (obviously, M is equal or larger than N), with look-up tables for the first phonetic mapping step, if the phonemes for one speaker voice are mapped onto those of a single voice for each  
20 foreign language, then N-1 different tables will have to be generated for each speaker voice, thus adding up to a total of  $N*(M-1)$  look-up tables.

In the case of a synthesis system operating with fifteen languages and two speaker voices for each  
25 language (which corresponds to a current arrangement adopted in the Loquendo TTS text-to-speech system developed by the Assignee of the instant application) then 435 look-up table would be required. That figure is quite significant, especially if one takes into  
30 account the possible requirement of generating such look-up tables manually.

Expanding such a system to include just one new speaker voice speaking one new language would require  $M+N=45$  new tables to be added. In that respect, one has  
35 to take into account that new phonemes are frequently

added to text-to-speech systems for one or more languages, this being a common case when the new phoneme added is an allophone of an already existing phoneme in the system. In that case, the need will  
5 exist of reviewing and modifying all those look-up tables pertaining to the language(s) to which the new phoneme is being added.

Object and summary of the invention

In view of the foregoing, the need exists for  
10 improved text-to-speech systems dispensing with the drawbacks of the prior art of the arrangements considered in the foregoing. More specifically, the object of the present invention is to provide a multi lingual text-to-speech system that:

- 15 - may dispense with the requirement of relying on multi-lingual speakers, and
- may be implemented by resorting to simple architectures, with moderate memory requirements, while also dispensing with the need of generating (possibly  
20 manually) a relevant number of look-up tables, especially when the system is improved with the addition of a new phoneme for one or more languages.

According to the present invention, that object is achieved by means of a method having the features set  
25 forth in the claims that follow. The invention also relates to a corresponding text-to-speech system and a computer program product loadable in the memory of at least one computer and comprising software code portions for performing the steps of the method of  
30 invention when the product is run on a computer. As used herein, reference to such a computer program product is intended to be equivalent to reference to a computer-readable medium containing instructions for controlling a computer system to coordinate the  
35 performance of the method of the invention. Reference

to "at least one computer" is evidently intended to highlight the possibility for the system of the invention to be implemented in a distributed fashion.

A preferred embodiment of the invention is thus an  
5 arrangement for the text-to-speech conversion of a text in a first language including sections in at least one second language, including:

- a grapheme/phoneme transcriptor for converting  
10 said sections in said second language into phonemes of said second language,

- a mapping module configured for mapping at least part of said phonemes of said second language onto sets of phonemes of said first language,

- a speech-synthesis module adapted to be fed with  
15 a resulting stream of phonemes including said sets of phonemes of said first language resulting from said mapping and the stream of phonemes of said first language representative of said text, and to generate a speech signal from said resulting stream of phonemes;  
20 the mapping module is configured for:

- carrying out similarity tests between each said phoneme of said second language being mapped and a set of candidate mapping phonemes of said first language,

- assigning respective scores to the results of  
25 said tests, and

- mapping said phoneme of said second language onto a set of mapping phonemes of said first language selected out of said candidate mapping phonemes as a function of said scores.

30 Preferably, the mapping module is configured for mapping said phoneme of said second language into a set of mapping phonemes of said first language selected out of:

- a set of phonemes of said first language including three, two or one phonemes of said first language, or

- an empty set, whereby no phoneme is included in  
5 said resulting stream for said phoneme in said second language.

Typically, mapping onto said empty set of phonemes of said first language occurs for those phonemes of said second language for which any of said scores fails  
10 to reach a threshold value.

The resulting stream of phonemes can thus be pronounced by means of a speaker voice of said first language.

Essentially, the arrangement described herein is  
15 based on a phonetic mapping arrangement wherein each of the speaker voices included in the system is capable of reading a multilingual text without modifying the vocalic database. Specifically, a preferred embodiment of the arrangement described herein seeks, among the  
20 phonemes present in the table for the language of the speaker voice, the phoneme that is most similar to the foreign language phoneme received as an input. The degree of similarity between the two phonemes can be expressed on the basis of phonetic-articulatory  
25 features as defined e.g. according to the international standard IPA. A phonetic mapping module quantifies the degree of affinity/similarity of the phonetic categories and the significance that each of them in the comparison between phonemes.

30 The arrangement described herein does not include any "acoustic" comparison between the segments included the database for the speaker voice language and the signal synthesized by means of the foreign language speaker voice. Consequently, the whole arrangement is  
35 less cumbersome from the computational viewpoint and

dispenses with the need for the system to have a speaker voice available for the "foreign" language: the sole grapheme-phoneme transcriptor will suffice.

Additionally, phonetic mapping is language  
5 independent. The comparison between phonemes refers exclusively to the vector of the phonetic features associated with each phoneme, these features being in fact language-independent. The mapping module is thus "unaware" of the languages involved, which means that  
10 no requirements exist for any specific activity to be carried out (possibly manually) for each language pair (or each voice-language pair) in the system. Additionally, incorporating new languages or new phonemes to the system will not require modifications  
15 in the phonetic mapping module.

Without losses in terms of effectiveness, the arrangement described herein leads to an appreciable simplification in comparison to prior art system, while also involving a higher degree of generalization with  
20 respect to previous solutions.

Experiments carried out show that the object of putting a monolingual speaker voice in a position to speak foreign languages in an intelligible way is fully met.

25 Brief description of the annexed drawings

The invention will now be described, by way of example only, by referring to the annexed figures of drawing, wherein:

- figure 1 is a block diagram of a text-to-speech  
30 system adapted to incorporate the improvement described herein, and

- figures 2 to 8 are flow charts exemplary of possible operation of the text-to-speech system of figure 1.



Detailed description of preferred embodiments of the invention

The block diagram of figure 1 depicts the overall architecture of a text-to-speech system of the multi  
5 lingual type.

Essentially, the system of figure 1 is adapted to receive as its input text that essentially qualifies as "multilingual" text.

Within the context of the invention, the  
10 significance of the definition "multilingual" is twofold:

- in the first place, the input text is multilingual in that it correspond to text written in any of a plurality of different languages T1,..., Tn such  
15 as e.g. fifteen different languages, and

- in the second place, each of the texts T1,..., Tn is per se multilingual in that it may include words or sentences in one or more languages different from the basic language of the text.

20 The text T1,..., Tn is supplied to the system (generally designated 10) in electronic text format.

Text originally available in different forms (e.g. as hard copies of a printed text) can be easily converted into an electronic format by resorting to  
25 techniques such as OCR scan reading. These methods are well known in the art, thus making it unnecessary to provide a detailed description herein.

A first block in the system 10 is represented by a language recognition module 20 adapted to recognize  
30 both the basic language of a text input to the system and the language(s) of any "foreign" words or sentences included in the basic text.

Again, modules adapted to perform automatically such a language-recognition function are well known in  
35 the art (e.g. from orthographic correctors of word

processing systems), thereby making it unnecessary to provide a detailed description herein.

In the following, in describing an exemplary embodiment of the invention reference will be made to a situation where the basic input text is an Italian text including words or short sentences in the English language. The speaker voice will also be assumed to be Italian.

Cascaded to the language-recognition module 20 are three modules 30, 40, and 50.

Specifically, module 30 is a grapheme/phoneme transcriptor adapted to segment the text received as an input into graphemes (e.g. letters or groups of letters) and convert it into a corresponding stream of phonemes. Module 30 may be any grapheme/phoneme transcriptor of a known type as included in the Loquendo TTS text-to-speech system already referred to in the foregoing.

Essentially, the output from the module 30 will be a stream of phonemes including phonemes in the basic language of the input text (e.g. Italian) having dispersed into it "bursts" of phonemes in the language(s) (e.g. English) comprising the foreign language words or short sentences included in the basic text.

Reference 40 designates a mapping module whose structure and operation will be detailed in the following. Essentially, the module 40 converts the mixed stream of phonemes output from the module 30 - comprising both phonemes of the basic language (Italian) of the input text as well as phonemes of the foreign language (English) - into a stream of phonemes including only phonemes of the first, basic language, namely Italian in the example considered.

Finally, module 50 is a speech-synthesis module adapted to generate from the stream of (Italian) phonemes output from the module 40 a synthesized speech signal to be fed to a loudspeaker 60 to generate a  
5 corresponding acoustic speech signal adapted to be perceived, listened to and understood by humans.

A speech signal synthesis module such as module 60 shown herein is a basic component of any text-to-speech signal, thus making it unnecessary to provide a  
10 detailed description herein.

The following is a description of operation of the module 40.

Essentially, the module 40 is comprised of a first and a second portion designated 40a and 40b,  
15 respectively.

The first portion 40a is configured essentially to pass on to the module 50 those phonemes that are already phonemes of the basic language (Italian, in the example considered).

20 The second portion 40b includes a table of the phonemes of the speaker voice (Italian) and receives as an input the stream of phonemes in a foreign language (English) that are to be mapped onto phonemes of the language of the speaker voice (Italian) in order to  
25 permit such a voice to pronounce them.

As indicated in the foregoing, the module 20 indicates to the module 40 when, within the framework of a text in a given language, a word or sentence in a foreign language appears. This occurs by means of a  
30 "signal switch" signal sent from the module 20 to the module 40 over a line 24.

Once again, it is recalled that reference to Italian and English as two languages involved in the text-to-speech conversion process is merely of an  
35 exemplary nature. In fact, a basic advantage of the

arrangement described herein lies in that phonetic mapping, as performed in portion 40b of the module 40 is language independent. The mapping module 40 is unaware of the languages involved, which means that no requirements exist for any specific activity to be carried out (possibly manually) for each language pair (or each voice-language pair) in the system.

Essentially, in the module 40 each "foreign" language phoneme is compared with all the phonemes present in the table (which may well include phonemes that - per se - are not phonemes of the basic language).

Consequently, to each input phoneme, a variable number of output phonemes may correspond: e.g. three phonemes, two phonemes, one phoneme or no phoneme at all.

For instance, a foreign diphthong will be compared with the diphthongs in the speaker voice as well as with vowel pairs.

A score is associated with each comparison performed.

The phonemes finally chosen will be those having the highest score and a value higher than a threshold value. If no phonemes in the speaker voice reach the threshold value, the foreign language phoneme will be mapped onto a nil phoneme and, therefore, no sound will be produced for that phoneme.

Each phoneme is defined in a univoque manner by a vector of n phonetic articulatory categories of variable lengths. The categories, defined according to the IPA standard, are the following:

- (a) the two basic categories vowel and consonant;
- (b) the category diphthong;

- (c) the vocalic (i.e. vowel) characteristics unstressed/stressed, non-syllabic, long, nasalized, rhoticized, rounded;
- (d) the vowel categories front, central, back;
- 5 - (e) the vowel categories close, close-close-mid, close-mid, mid, open-mid, open-open-mid, open;
- (f) the consonant mode categories plosive, nasal, trill, tapflap, fricative, lateral-fricative, approximant, lateral, affricate;
- 10 - (g) the consonant place categories bilabial, labiodental, dental, alveolar, postalveolar, retroflex, palatal, velar, uvular, pharyngeal, glottal; and
- (h) the other consonant categories voiced, long, syllabic, aspirated, unreleased, voiceless,
- 15 semiconsonant.

In actual fact, the category "semiconsonant" is not a standard IPA feature. This category is a redundant category used for the simplicity of notation to denote an approximate/alveolar/palatal consonant or  
20 an approximant-velar consonant.

The categories (d) and (e) also describe the second component of a diphthong.

Each vector contains one category (a), one or none category (b) if the phoneme is a vocal, at least one  
25 category (c) if the phoneme is a vocal, one category (d) if the phoneme is a vocal, one category (e) if the phoneme is a vocal, one category (f) if the phoneme is a consonant, at least one category (g) if the phoneme is a consonant and at least one category (h) if the  
30 phoneme is a consonant.

The comparison between phonemes is carried out by comparing the corresponding vectors, allotting respective scores to said vector-by-vector comparisons.

The comparison between vectors is carried out by  
35 comparing the corresponding categories, allotting

respective score values to said category-by-category comparisons, said respective score values being aggregate to generate said scores.

Each category-by-category comparison has  
5 associated a differentiated weight, so that different category-by-category comparisons can have different weights in generating the corresponding score.

For example, a maximum score value obtained comparing (f) categories will be always lower than the  
10 score value obtained comparing (g) categories (i.e. the weight associated to category (f) comparison is higher than the weight associated to category (g) comparison). As a consequence, the affinity between vectors (score) will be influenced mostly by the similarity between  
15 categories (f), compared with the similarity between categories (g).

The process described in the following uses a set of constants having preferably the following values;

- *MaxCount* = 100  
20 - *Kopen* = 14  
- *Sstep* = 1  
- *Mstep* = 2\* *Lstep*  
- *Lstep* = 4\* *Mstep*  
- *Kmode* = *Kopen* + (*Lstep* \* 2)  
25 - *Thr* = *Kmode*  
- *Kplace3* = 1  
- *Kplace2* = (*Kplace3* \* 2) + 1  
- *Kplace1* = ((*Kplace2*) \* 2) +1  
- *DecrOPen* = 5

30 Operation of the system exemplified herein will now be described by referring to the flow charts of figures 2 to 8 by assuming that a single phoneme is brought to the input of the module 40. If a plurality of phonemes are supplied as an input to the module 40,

the process described in the following will be repeated for each input phoneme.

In the following a phoneme having the category diphthong or affricate will be designated "divisible phoneme".

When defining the mode and place categories of a phoneme, these are intended to be univocal unless specified differently.

For instance if a given foreign phoneme (e.g. PhonA) is termed fricative - uvular, this means that it has a single mode category (fricative) and a single place category (uvular).

By referring first to the flow chart of figure 2 in a step 100 an index (Indx) scanning a table of the speaker voice language (hereinafter designated TabB) is set to zero, namely positioned at the first phoneme in the table.

The score value (Score) is set to zero initial value as is the case of the variables MaxScore, TmpScrMax, FirstMaxScore, Loop and Continue. The phonemes BestPhon, FirstBest and FirstBestCmp are set at the nil phoneme.

In a step 104 the vector of the categories for the foreign phoneme (PhonA) is compared with the vector of the phoneme for a speaker voice language (PhonB).

If the two vectors are identical, the two phonemes are identical and in a step 108 the score (Score) is adjourned to the value MaxCount and the subsequent step is a step 144.

If the vectors are different, in a step 112 the base categories (a) are compared.

Three alternatives exist: both phonemes are consonants (128), both are vowels (116) or different (140).

In the step 116 a check is made as to whether PhonA is a diphthong. In the positive, in a step 124 the functions described in the flow chart of figure 4 are activated as better detailed in the following.

5        If it is not a diphthong, in a step 120, the function described in the flow chart of figure 5 is activated in order to compare a vowel with a vowel.

It will be appreciated that both steps 120 and 124 may lead to the score being modified as better detailed  
10    in the following.

Subsequently, processing evolves towards the step 144.

In a step 128 (comparison between consonants) a check is made as to whether PhonA is affricate. In the  
15    positive, in a step 136 the function described in the flow chart of figure 7 is activated. Alternatively, in a step 132 the function described in figure 6 is activated in order to compare the two consonants.

In a step 140 the functions described in the flowchart of figure 8 are activated as better detailed  
20    in the following.

Similarly better detailed in the following are theos criteria based on which the score may be modified in both steps 132 and 136.

25        Subsequently, the system evolves towards the step 144.

The results of comparison converge towards the step 144 where the score value (Score) is read.

In a step 148, the score value is compared with a  
30    value designated MaxCount. If the score value equals MaxCount the search is terminated, which means that a corresponding phoneme in a speaker voice language has been found for PhonA (step 152).



If the score value is lower than MaxCount (which is checked in a step 148), in a step 156 processing proceeds as described in the flow chart of figure 3.

In a step 160, the value Continue is compared with the value 1. In the positive (namely Continue equals 1), the system evolves back to step 104 after setting the value Loop to the value 1 and resetting Continue, Indx and Score to zero values. Alternatively, the system evolves towards the step 164.

From here, if PhonA is nasalized or rhoticized and the phoneme or the phonemes selected are not either of these kinds, the system evolves towards the step 168, where the phoneme/s selected is supplemented by a consonant from TabB whose phonetic-articulatory characteristics permit to simulate the nasalized or the rhoticized sound of PhonA.

In a step 172, the phoneme (or the phonemes) selected are sent towards the output phonetic mapping module 40 to be supplied to the module 50.

The step 200 of figure 3 is reached from the step 156 of the flow chart of figure 2.

From the step 200, the system evolves towards a step 224 if one of the two conditions is met:

- PhonA is a diphthong to be mapped onto two vowels;

- PhonA is affricate, PhonB is non-affricate consonant but may be the component of an affricate.

The parameter Loop indicates how many times the table TabB has been scanned from top to bottom. Its value may be 0 or 1.

Loop will be set to the value 1 only if PhonA is diphtong or affricate, whereby it is not possible to reach a step 204 with Loop equal to 1. In the step 204 the Maximum Condition is checked. This is a met if the score value (Score) is higher than MaxScore or if is

equal thereto and the set of n phonetic features for PhonB is shorter than the set for BestPhon.

If the condition is met, the system evolves towards a step 208 where MaxScore is adjourned to the score value and PhonB becomes BestPhon.

In a step 212 Indx is compared with TabLen (the number of phonemes in TabB).

If Indx is higher than or equal to TabLen, the system evolves towards a step 284 to be described in the following.

If Indx is lower, then PhonB is not the last phoneme in the table and the system evolves towards a step 220, wherein Indx is increased by 1.

If PhonB is the last phoneme in the table, then the search is terminated and BestPhon (having associated the score MaxScore) is the candidate phoneme to substitute PhonA.

In a step 224 the value for Loop is checked.

If Loop is equal to 0, then the system evolves towards a step 228 where a check is made as to whether PhonB is diphthong or affricate.

In the positive (i.e. if PhonB is diphthong or affricate), the subsequent step is a step 232.

At this point, in a step 232 the Maximum Condition is checked between Score and MaxScore.

If the condition is met (i.e. Score is higher than MaxScore), in a step 236 MaxScore is adjourned to the value of Score and the PhonB becomes BestPhon.

In a step 240 (which is reached if the check of the step 228 shows that PhonB is neither diphthong nor affricate), a check is made as to whether a maximum condition exists between Score and TmpScrMAX (with the FirstBestComp in the place of BestPhon). If this is satisfied (i.e. Score is higher than TmpScrMAX), in a

step 244 TmpScrMax is adjourned by means of Score and FirstBestComp by means of PhonB.

In a step 248, a check is made as to whether PhonB is the last phoneme in TabB (then Indx is equal to  
5 TabLen).

In the positive (252), the value for MaxScore is stored as the variable FirstMaxScore, BestPhon is stored as a FirstBest and subsequently, in a step 256, Indx is set to 0, while Continue is set to 1 (so that  
10 also the second component for PhonA will be searched), and Score is set to 0.

A step 260 is reached from the step 224 if Loop is equal to 1, namely if PhonB is scrutinized as a possible second component for PhonA. In a step 260, a  
15 check is made as to whether the maximum condition is satisfied in the comparison between Score and MaxScore (which pertains to BestPhon).

In a step 264, Score is stored in MaxScore and PhonB in BestPhon in the case the maximum condition is  
20 satisfied. In a step 268 a check is made as to whether PhonB is the last phoneme in the table and, in the positive, the system evolves towards the step 272.

In the step 272, a phoneme most similar to PhonA can be selected between a divisible phoneme or a couple  
25 of phonemes in the speaker language voice depending on whether the condition FirstMaxScore larger or equal than (TmpScrMax + MaxScore) is satisfied. The higher value of the two members of the relationship is stored as a MaxScore. In the case the choice falls on a pair  
30 of phonemes, this will be FirstBestCmp and BestPhon. Otherwise only FirstBest will be considered.

It is worth pointing out that BestPhon (found at the second iteration) cannot be diphthong or affricate. In a step 276, Indx is increased by 1 and Score is set  
35 to 0.

From the step 280 the system evolves back to the step 104.

The step 284 is reached from the step 272 (or the step 212) when the search is completed. In the step 284  
5 a comparison is made between MaxScore and a threshold constant Thr. If MaxScore is higher, then the candidate phoneme (or the phoneme pair) is the substitute for PhonA. In the negative, PhonA is mapped onto the nil phoneme.

10 The flow chart of the figure 4 is a detailed description of the block 124 of the diagram of figure 2.

A step 300 is reached if PhonA is a diphthong.

In a step 302 a check is made as to whether PhonB  
15 is a diphthong and Loop is equal to 0. In the positive, the system evolves towards the step 304 where, after checking the features for PhonA, the system evolves towards a step 306 if PhonA is a diphthong to be mapped onto a single vowel.

20 The diphthongs of this type have a first component that is mid and central and the second component that is close-close-mid and back.

From the step 306 the system evolves towards the step 144.

25 In a step 308, the function comparing two diphthongs is called.

In a step 310, the categories (b) of the two phonemes are compared via that function and Score is increased by 1 for each common feature found.

30 In a step 312, the first components of the two diphthongs are compared and in a step 314 a function called F\_CasiSpec\_Voc is called for the two components.

This function performs three checks that are satisfied if:

- the components of the two diphthongs are indistinctly vowel open, or vowel open-open-mid, front and not rounded, or open-mid, back and not rounded;

- the component of PhonA is mid and central, and  
5 in TabB no phonemes exist exhibiting both categories, and PhonB is close-mid and front;

- the component of PhonA is close, front and rounded, or close-close-mid, front and rounded, and in TabB no phonemes exist having such features while PhonB  
10 is close, back, and rounded or close-close-mid, back and rounded.

If any of the three conditions is met, in a step 316 the value for Score is adjourned by adding (KOpen \* 2) thereto.

15 Otherwise, in a step 318, a function F\_ValPlace\_Voc is called for the two components.

Such a function compares the categories front, central and back (categories (d)).

If identical, Score is incremented by Kopen; if  
20 they are different, a value is added to Score which is comprised of KOpen minus the constant DecrOpen if the distance between the two categories is 1, while Score is not incremented if the distance is 2.

A distance equal to one exists between central and  
25 front and between central and back, while a distance equal to two exists between front and back.

In step 320 a function F\_ValOpen\_Voc is called for comparing the two components of the diphthong. Specifically, F\_ValOpen\_Voc operates in cyclical manner  
30 by comparing the first components and the second components in two subsequent iterations.

The function compares the categories (e) and adds to Score the constant KOpen less the value of the distance between the categories as reported in Table 1  
35 hereinafter.

The matrix is symmetric, whereby only the upper portion was reported.

By making a numerical example, if PhonA is a close vowel and PhonB is a close-mid vowel, a value equal to  
5 (KOpen-(6 \* Lstep)) will be added to Score which, by considering the value of the constants, is equal to 8.

In a step 322, if the components have both the rounded feature, the constant (KOpen + 1) is added to Score. Conversely, if only one of the two is rounded,  
10 then Score is decremented by KOpen.

From the step 324 the system goes back to the step 314 if the two first components have been compared; conversely, a step 326 is reached when also the second components have been compared.

15 In the step 326, the comparison of the two diphthongs is terminated and the system evolves back to the step 144.

In a step 328 a check is made as to whether PhonB is a diphthong and Loop is equal to 1. If that is the  
20 case, the system evolves towards a step 306.

In a step 330, a check is made as to whether PhonA is a diphthong to be mapped onto a single vowel. If that is the case, in a step 331 Loop is checked and, if found equal to 1, the step 306 is reached.

25 In a step 332, a phoneme TmpPhonA is created.

TmpPhonA is a vowel without the diphthong characteristic and having close-mid, back and rounded features.

Subsequently, the system evolves to a step 334  
30 where the TmpPhonA and PhonB are compared. The comparison is effected by calling the comparison function between two vowel phonemes without the diphthong category.

That function, which is called also at the step 120 in the flow chart of figure 2, is described in detail in figure 5.

In a step 336, the function is called to perform a  
5 comparison between a component of PhonA and PhonB:  
consequently, in a step 338, if Loop is equal to 0, the  
first component of PhonA is compared with PhonB (in a  
step 344). Conversely, if Loop is equal to 1, the  
second component of PhonA is compared with PhonB (in a  
10 step 340).

In the step 340, reference is made to the  
categories nasalized and rhoticized, by increasing  
Score by one for each identity found.

In a step 342, if PhonA bears a stress on its  
15 first component and PhonB is a stressed vowel, or if  
PhonA is unstressed or bears a stress on its second  
component and PhonB is an unstressed vowel, Score is  
incremented by 2. In all other cases it is decreased by  
2.

In a step 344, if PhonA bears its stress on the  
20 second component and PhonB is a stressed vowel, or if  
PhonA is stressed on the first consonant or is an  
unstressed diphthong and PhonB is an unstressed vowel,  
then Score is increased by 2; conversely, it is  
25 decreased by 2 in all other cases.

In 348, the categories (d) and (e) of the first or  
second component of PhonA (depending on whether Loop is  
equal to 0 or 1, respectively) are compared with PhonB.

Comparison of the feature vectors and updating  
30 Score is performed based on the same principles already  
described in connection with the steps from 314 to 322.

A step 350 marks the return to step 144.

The flow chart of figure 5 describes in detail the  
step 120 of the diagram of figure 2, namely the  
35 comparison between two vowels that are not diphthongs.

In a step 400 a check is made as to whether PhonB is a diphthong. In the positive, the system evolves directly towards a step 470.

In a step 410, a comparison is made based on the  
5 categories (b) by increasing Score by 1 for each category found to be identical.

Conversely, in a step 420, the function F\_CasiSpec\_Voc already described in the foregoing is called in order to check whether one of the conditions  
10 of the function is met.

If that is the case, Score is increased by the quantity ( $K_{Open} * 2$ ) in a step 430.

In the case of a negative outcome, in a step 440 function F\_ValPlace\_Voc is called.

Subsequently, in a step 450, the function F\_ValOpen\_Voc is called.  
15

In a step 460, if both vowels have the rounding category, Score is increased by the constant ( $K_{Open} + 1$ ); if, conversely, only one phoneme is found to have  
20 the rounded category, then Score is decremented by  $K_{Open}$ .

A step 470 marks the end of the comparison, after which the system evolves back to the step 144.

The flow chart of figure 6 describes in detail the  
25 block 132 in the diagram of figure 1.

In a step 500 the two consonants are compared, while the variable TmpKP is set to 0 and the function F\_CasiSpec\_Cons is called in a step 504.

The function in question checks whether any of the  
30 following conditions are met;

1.0 PhonA uvular-fricative and in TabB there are no phonemes with these characteristics and PhonB is trill-alveolar;

1.1 PhonA uvular fricative and in TabB there are no  
35 phonemes with these characteristics



PhonB is approximant-alveolar;

1.2 PhonA uvular fricative and in TabB there are no phonemes with these characteristics and

PhonB is uvular-trill;

5 1.3 PhonA uvular fricative and in TabB there are no phonemes with these characteristics or with those of PhonB of 1.0 or 1.1 or 1.2, and PhonB is lateral-alveolar;

2.0 PhonA glottal fricative and in TabB there are no phonemes with these characteristics and

PhonB is fricative-velar;

3.0 PhonA fricative-velar and in TabB there are no phonemes with these characteristics and

PhonB is fricative-glottal or plosive-velar;

15 4.0 PhonA trill-alveolar and in TabB there are no phonemes with these characteristics

and PhonB is fricative-uvular;

4.1 PhonA trill-alveolar and in TabB there are no phonemes with these characteristics

20 and PhonB is approximant-alveolar;

4.2 PhonA trill-alveolar and in TabB there are no phonemes with these characteristics or with those of PhonB of 4.0 and 4.1, and PhonB is lateral-alveolar;

25 5.0 PhonA nasalized-velar and in TabB there are no phonemes with these characteristics and

PhonB is nasalized-alveolar;

5.1 PhonA nasalized-velar and in TabB there are no phonemes with these characteristics or with those of

30 PhonB of 5.0 and PhonB is nasalized-bilabial;

6.0 PhonA is fricative-dental-non voiced and in TabB there are no phonemes with these characteristics and PhonB is approximant-dental;

6.1 PhonA is fricative-dental-non voiced and in TabB  
35 there are no phonemes with these characteristics or

with those of PhonB of 6.0, and PhonB is plosive-dental;

6.2 PhonA is fricative-dental-non voiced and in TabB there are no phonemes with these characteristics or

5 those of PhonB of 6.0 and PhonB is plosive-alveolar;

7.0 PhonA is fricative-dental-voiced and in TabB there are no phonemes with these characteristics and PhonB is approximant-dental;

7.1 PhonA is fricative-dental-voiced and in TabB there are no phonemes with these characteristics or those of PhonB of 7.0 and PhonB is plosive-dental;

7.2 PhonA is fricative-dental-voiced and in TabB there are no phonemes with these characteristics or those of PhonB of 7.0 and PhonB is plosive-alveolar;

15 8.0 PhonA is fricative-palatal-alveolar-non voiced and in TabB there are no phonemes with these characteristics and PhonB is fricative-postalveolar;

8.1 PhonA is fricative-palatal-alveolar-non voiced and in TabB there are no phonemes with these

20 characteristics or those of PhonB of 8.0 and PhonB is fricative-palatal;

9.0 PhonA is fricative-postalveolar e in TabB there are no phonemes with these characteristics or fricative-retroflex and PhonB is fricative-alveolar-palatal;

25 10.0 PhonA is fricative-postalveolar-velar and in TabB there are no phonemes with these characteristics and PhonB is fricative-alveolar-palatal;

10.1 PhonA is fricative-postalveolar-velar and in TabB there are no phonemes with these characteristics and PhonB is fricative -palatal;

30 10.2 PhonA is fricative-postalveolar-velar and in TabB there are no phonemes with these characteristics or those of 10.0 or 10.1 and PhonB is fricative-postalveolar;

11.0 PhonA is plosive-palatal and in TabB there are no phonemes with these characteristics and PhonB is lateral-palatal;

11.1 PhonA is plosive-palatal and in TabB there are no  
5 phonemes with these characteristics or those of PhonB di 11.0 and PhonB is fricative-palatal or approximant-palatal;

12.0 PhonA is fricative-bilabial-dental-voiced and in TabB there are no phonemes with these characteristics  
10 and PhonB is approximant-bilabial-voiced;

13.0 PhonA is fricative-palatal-voiced and in TabB there are no phonemes with these characteristics and PhonB is plosive-palatal-voiced or approximant-palatal-voiced;

14.0 PhonA is lateral-palatal and in TabB there are no phonemes with these characteristics and PhonB is plosive-palatal;

14.1 PhonA is lateral-palatal and in TabB there are no phonemes with these characteristics or those of PhonB  
20 of 14.0 and PhonB is fricative-palatal or approximant-palatal;

15.0 PhonA is approximant-dental and in TabB there are no phonemes with these characteristics and PhonB is plosive-dental or plosive-alveolar;

16.0 PhonA is approximant-bilabial and in TabB there are no phonemes with these characteristics and PhonB is plosive-bilabial;

17.0 PhonA is approximant-velar and in TabB there are no phonemes with these characteristics and PhonB is  
30 plosive-velar;

18.0 PhonA is approximant-alveolar and in TabB there are no phonemes with these characteristics and PhonB is trill-alveolar or fricative-uvular o trill-uvular;

18.1 PhonA is approximant-alveolar and in TabB there are no phonemes with these characteristics or those of PhonB in 18.0 and PhonB is lateral-alveolar.

If any of these conditions is met, the system evolves towards a step 508 where TmpPhonB is substituted for PhonB during the whole process of comparison up to a step 552.

If none of the conditions above is met, the system evolves directly towards a step 512 where the mode categories (f) are compared.

If PhonA and PhonB have the same category, then Score is increased by KMode.

In a step 516 a function F\_CompPen\_Cons is called to control if the following condition is met:

- PhonA is fricative-postalveolar and PhonB (or TmpPhonB) is fricative-postalveolar-velar.

If the condition is met, then Score is decreased by KPlace1.

In a step 520 a function F\_ValPlace\_Cons is called to increment TmpKP based on what is reported in Table 2.

In the table in question the categories for PhonA are on the vertical axis and those for PhonB on the horizontal axis. Each cell includes a bonus value to be added to Score.

By assuming, by way of example, that PhonA has the category labiodental and PhonB the dental category only, then, by scanning the line for labiodental, and crossing the column for dental, one finds that the value Kplace2 will have to be added to Score.

In a step 524, a check is made as to whether PhonA is approximant-semivowel and PhonB (or TmpPhonB) is approximant. If the check yields a positive result, the system evolves towards a step 528, where a test is made on TmpKP.

Such a test is made in order to ensure that in the case the two phonemes being compared are both approximant and with identical place categories, their Score is higher than in the case of any comparison  
5 consonant-vocal.

If such a variable is larger or equal to KPlace1, then in a step 532 TmpKP is increased by KMode. In the negative, TmpKP is set to zero in a step 536.

In a step 540 the quantity TmpKP is added to  
10 Score.

In a step 544 a check is made as to whether Score is higher then KMode.

If that is the case, in a step 548 the categories (h) are compared with the exception of the  
15 semiconsonant category. For each identity found, Score is increased by one.

A step 552 marks the end of the comparison, after which the system evolves back to step 144 of figure 1.

The flow chart of figure 7 refers to the  
20 comparison between phonemes in the case PhonA is an affricate consonant (step 136 of figure 2).

In a step 600 the comparison is started and in a step 604 a check is made as to whether PhonB is affricate and Loop is equal to 0.

25 If that is the case, the system evolves towards a step 608, which in turn causes the system to evolve back to step 132.

In a step 612, a check is made as to whether PhonB is affricate and Loop is equal to 1.

30 If that is the case, a step 660 is directly reached.

In a step 616, a check is made as to whether PhonB can be considered as comprised of an affricate.

This cannot be the case if Loop is equal to 1 and  
35 PhonB has the categories fricative-postsalveolar-velar.

If that is the case, the system evolves to wards step 660.

In a step 620, a check is made for the value of Loop: if that is equal to 0, the system evolves towards  
5 a step 642.

In that step, PhonA is temporarily substituted in the comparison with PhonB by TmpPhonA; this has the same characteristics of PhonA, but for the fact that in the place of being affricate it is plosive.

10 In a step 628, a check is made as to whether TmpPhonA has the labiodental categories; if that is the case in a step 636, the dental categories removed from the vector of categories.

In a step 632, a check is made as to whether  
15 TmpPhonA has the postalveolar category; in the positive, such category is replaced in a step 644 by the alveolar category.

In a step 640, a check is made as to whether TmpPhonA has the categories alveolar-palatal; if that  
20 is the case the palatal category is removed.

In a step 652 phonA is temporarily replaced (until reaching the step 144) in comparison with PhonB by TmpPhonA; this has the same characteristics of PhonA, but for the fact that it is fricative in the place of  
25 being affricate.

A step 656 marks the evolution towards the comparison of the step 132 by comparing TmpPhonA with PhonB.

A step 660 marks the return to step 144.

30 The flow chart of figure 8 describes in detail the step 140 of the flow chart of figure 2.

A step 700 is reached if PhonA is consonant and PhonB is vowel or if PhonA is vowel and PhonB is consonant. The phoneme TmpPhonA is set as the nil  
35 phoneme.

In a step 705, a check is made as to whether PhonA is vowel and PhonB is consonant. In the positive the next step is step 780

In a step 710, a check is made as to whether PhonA  
5 is approximant-semiconsonant.

In the negative, the system evolves directly to a step 780.

In a step 720, a check is made as to whether PhonA is palatal. If that is the case, in a step 730 TmpPhonA  
10 is transformed into a unstressed-front-close vowel and the comparison of a step 120 is performed between TmpPhonA and PhonB.

In a step 740, a check is made as to whether PhonA is bilabial-velar. If that is the case, in a step 750  
15 TmpPhonA is transformed into an unstressed-close-back-rounded vowel and the comparison of the step 120 (figure 2) is performed between TmpPhonA and PhonB.

In a step 760, a check is made as to whether PhonA is bilabial -palatal. If that is the case, in a step  
20 770 TmpPhonA is transformed into an unstressed-close-back-rounded vowel and the comparison of the step 120 is carried out between TmpPhonA and PhonB.

A step 780 marks the evolution of the system back to the step 144.

25 In the following the two tables 1 and 2 repeatedly referred in the foregoing are reported.

	CLOSE	CLOSE-CLOSE-MID	CLOSE-MID	MID	OPEN-MID	OPEN-OPEN-MID	OPEN
CLOSE	0	2*LStep	6*LStep	7*LStep	8*LStep	12*LStep	14*LStep
CLOSE-CLOSE-MID		0	4*LStep	5*LStep	6*LStep	10*LStep	12*LStep
CLOSE-MID			0	1*LStep	2*LStep	6*LStep	8*LStep
MID				0	1*LStep	5*LStep	7*LStep
OPEN-MID					0	4*LStep	6*LStep
OPEN-OPEN-MID						0	2LStep
OPEN							0

Table 1: Distances of vowel features (e)



	BILABIAL	LABIODENTAL	DENTAL	ALVEOLAR	POST ALVEOLAR	RETROFLEX	PALATAL	VELAR	UVULAR	PHARYNGEAL	GLOTTAL
BILABIAL	+KPlace1	+KPlace2	+0	+0	+0	+0	+0	+0	+0	+0	+0
LABIODENTAL	+KPlace2	+KPlace1	+KPlace2	+0	+0	+0	+0	+0	+0	+0	+0
DENTAL	+0	+0	+KPlace1	+KPlace2	+0	+0	+0	+0	+0	+0	+0
ALVEOLAR	+0	+0	+KPlace3	+KPlace1	+KPlace2	+KPlace3	+0	+0	+0	+0	+0
POSTALVEOLAR	+0	+0	+0	+KPlace3	+KPlace1	+KPlace2	+0	+0	+0	+0	+0
RETROFLEX	+0	+0	+0	+KPlace3	+KPlace3	+KPlace1	+KPlace2	+0	+0	+0	+0
PALATAL	+0	+0	+0	+0	+KPlace3	+KPlace2	+KPlace1	+KPlace2	+0	+0	+0
VELAR	+0	+0	+0	+0	+0	+0	+0	+KPlace1	+0	+0	+0
UVULAR	+0	+0	+0	+KPlace2	+0	+0	+0	+KPlace2	+KPlace1	+0	+0
PHARYNGEAL	+0	+0	+0	+0	+0	+0	+0	+0	+0	+KPlace1	+0
GLOTTAL	+0	+0	+0	+0	+0	+0	+0	+0	+0	+0	+KPlace 1

Table 2: values to be added to score

Of course, without prejudice to the underlying principles of the invention, the variance and embodiments may vary, also significantly, with respect  
5 to what has been described, by way of example only, without departing from the scope of the invention as defined by the annexed claims.